LA-UR-01-2894

Proceedings of
2001 ASME International Mechanical Engineering Congress and Exposition
November 11–16, 2001, New York

IMECE2001/DSC-10A-3

NOVELTY DETECTION USING AUTO-ASSOCIATIVE NEURAL NETWORK

Hoon Sohn
ESA-EA, MS C926
Los Alamos National Laboratory
Los Alamos, NM 87545
USA

Keith Worden
Dept. of Mechanical Engineering
University of Sheffield
Mappin Street, Sheffield S1 3JD
United Kingdom

Charles R. Farrar
ESA-EA, MS C946
Los Alamos National Laboratory
Los Alamos, NM 87545
USA

ABSTRACT
The primary objective of novelty detection is to examine if a system significantly deviates from the initial baseline condition of the system. In reality, the system is often subject to changing environmental and operation conditions affecting its dynamic characteristics. Such variations include changes in loading, boundary conditions, temperature, and humidity. Most damage diagnosis techniques, however, generally neglect the effects of these changing ambient conditions. Here, a novelty detection technique is developed explicitly taking into account these natural variations of the system in order to minimize false positive indications of true system changes. Auto-associative neural networks are employed to discriminate system changes of interest such as structural deterioration and damage from the natural variations of the system.

1. INTRODUCTION

Damage identification is a problem, which can be addressed at many levels. Stated in its most basic form, the objective is to ascertain simply if damage is present or not. One class of algorithms, which show considerable promise for this purpose, is grouped under the name *novelty detection methods*. The philosophy is simple; during the normal operation of a system or structure, measurement features are collected which characterize the normal conditions. After training the diagnostic in question, subsequent data can be examined to see if the features deviate significantly from the norm. That is, novelty detection is a technique for deciding if measurements from a system or structure indicate departure form previously established normal conditions. An alarm is signaled if the index value increased above a pre-determined threshold.

Unfortunately, matters are seldom as simple as this. In reality, structures will be subjected to changing environmental and operational states such as varying temperature, humidity, and loading conditions affecting the measured features and the normal condition. In this case, there may be a continuous range of normal conditions, and it is clearly undesirable for the novelty detector to signal damage simply because of a change in the environment or operation. In fact, these changes can often mask more subtle structural changes caused by damage.

One approach to solving this problem is to measure parameters related to these environmental and operational conditions as well as the vibration features over a wide range of these varying conditions to characterize the normal conditions. The normal conditions can be parameterized at different environmental and operational states. Then, a novelty detector, which does not provide false indication of damage under changing environmental and operational conditions, can be built. On the other hand, there are other cases where it is practically difficult to measure parameters related to the environmental and/or operational conditions. This paper addresses the later cases where no measurements are available for these natural variations.

The idea is based on auto-associative neural networks where target outputs are simply inputs to the network. Using the measured features corresponding to the normal conditions, the auto-associative neural network is trained to characterize the underlying dependency of the measured features on the unmeasured environmental and operational variations by treating these environmental and operational conditions as hidden intrinsic variables in the neural network.

The layout of this paper is as follows. In Section 2, a brief description of auto-associative neural network is given relating this network with Principal Component Analysis (PCA) and Nonlinear Principal Component Analysis (NLPCA). A measure of novelty or novelty index is defined in Section 3 using the auto-associative network outputs. In Section 4, the applicability of the auto-associative neural network to damage diagnosis problems is demonstrated on synthetic data sets obtained from a simplified model of a computer hard disk. The paper concludes with a general discussion in Section 5.

2. AUTO-ASSOCIATIVE NEURAL NETWORKS

PCA has been proven to facilitate many types of multivariate data analysis including data reduction and visualization, data validation, fault detection, and correlation analysis (Fukunaga and Koontz, 1970). Similar to PCA, NLPCA is used as an aid to multivariate data analysis. While PCA is restricted on mapping only linear correlations among variables, NLPCA can reveal the nonlinear correlations presented in data. If nonlinear correlations exist among variables in the original data, NLPCA can reproduce the original data with greater accuracy and/or with fewer factors than PCA. This NLPCA can be realized by training a feedforward neural network to perform the identity mapping, where the network outputs are simply the reproduction of network inputs. For this reason, this special kind of neural network is named as an *auto-associative neural network* (See Figure 1). The network consists of an internal "bottleneck" layer and two additional hidden layers. The bottleneck layer contains fewer nodes than input or output layers forcing the network to develop a compact representation of the input data. The NLPCA presented in this paper is a general purpose feature extraction/data reduction algorithm discovering features that contain the maximum amount of information from the original data set. In the following sections, PCA and NLPCA are briefly reviewed. More detailed discussions on PCA, NLPCA, and auto-associative networks can be found from Fukunaga (1990), Kramer (1991), Rumelhart and McClelland (1988), respectively.

2.1. Principal Component Analysis (PCA)

PCA is a linear transformation mapping multidimensional data into lower dimensions with minimum loss of information. Let **Y** represent the original data with the size of $m \times n$. Here, *m* is the number of variables and *n* is the number of data sets. PCA can be viewed as a linear mapping of data from the original dimension *m* to a lower dimension *d*:

$$\mathbf{X} = \mathbf{T}\mathbf{Y} \tag{1}$$

where **X** ($\in \Re^{d \times n}$) is called the *scores* matrix. **T** ($\in \Re^{d \times m}$) is called the *loading* matrix and $\mathbf{T}\mathbf{T}^{\mathrm{T}} = \mathbf{I}$. The loss of information in this mapping can be assessed by re-mapping the projected data back to the original space:

$$\hat{\mathbf{Y}} = \mathbf{T}^{\mathrm{T}}\mathbf{X} \tag{2}$$

Then, the reconstruction error (residual error) matrix E is defined as:

$$\mathbf{E} = \mathbf{Y} - \hat{\mathbf{Y}} \tag{3}$$

The smaller the dimension of the projected space, the greater the resulting error. The loading matrix **T** can be found such that the Euclidean norm of the residual matrix, ‖**E**‖, is minimized for the given size of *d*. It can be shown that the columns of **T** are the eigenvectors corresponding to the *d* largest eigenvalues of the covariance matrix of Y (Fukunaga, 1990).

2.2. Nonlinear Principal Component Analysis (NLPCA)

NLPCA generalizes the linear mapping by allowing arbitrary nonlinear functionalities. Similar to Equation (1), NLPCA seeks a mapping in the following form:

$$\mathbf{X} = \mathbf{G}(\mathbf{Y}) \tag{4}$$

where **G** is a nonlinear vector function and consists of *d* number of individual nonlinear functions: $\mathbf{G} = \{G_1, G_2, ..., G_d\}$. By analogy to Equation (2), the inverse transformation, restoring the original dimensionality of the data, is implemented by a second nonlinear vector function **H**:

$$\hat{\mathbf{Y}} = \mathbf{H}(\mathbf{X}) \tag{5}$$

The information lost is again measured by $\mathbf{E} = \mathbf{Y} - \hat{\mathbf{Y}}$. Similar to PCA, **G** and **H** are computed to minimize the Euclidean norm of ‖**E**‖ meaning minimum information loss in the same sense as PCA. NLPCA employs artificial neural networks to generate arbitrary nonlinear functions. Cybenko (1989) has shown that functions of the following form are capable of fitting any nonlinear function $\mathbf{y} = f(\mathbf{x})$ to an arbitrary degree of precision:

$$y_k = \sum_{j=1}^{N_2} w_{jk}^2 \sigma\left(\sum_{i=1}^{N_1} w_{ij}^1 x_i + b_j \right) \tag{6}$$

where y_k and x_i are the *k*th and *i*th components of **y** and **x**, respectively. w_{ij}^k represents the weight connecting the *i*th node in the *k*th layer to the *j*th node in the *(k+1)*th layer, and b_j is a node bias. $\sigma(x)$ is a monotonically increasing continuous function with the output range of 0 to 1 for an arbitrary input *x*. A sigmoid transfer function is often used in neural networks to realize this function.

Note that, to fit arbitrary nonlinear functions, at least two layers of weighted connections are required, and the first hidden layer should be composed of sigmoidal functions. Therefore, the two nonlinear vector functions in Equations (4) and (5) should have the same architecture: one hidden layer with sigmoidal functions and one output layer. The output layer can have either linear or sigmoidal transfer functions without affecting the generality of the mapping. For instance, the first hidden layer of **G**, which consists of M_1 nodes with sigmoidal functions, operates on the columns of **Y** mapping *m* inputs to M_1 node outputs. The output of the first hidden layer is projected into the bottleneck layer, which contains *d* nodes. In a similar fashion, the inverse mapping function **H** takes the columns of **X** as inputs relating *d* inputs to M_2 node outputs. The final output layer reconstructs the target output $\hat{\mathbf{Y}}$, and contains *m* nodes. This network architecture consisted of mapping and de-mapping **G** and **H** is shown in Figure 1. It should be noted that if the neural networks for **G** and **H** are to be trained separately, the target output **X** is unknown for the training of the **G** network. For the same reason, the input for the **H** network is not known. It is observed that **X** is both the

output of **G** and the input of **H**. Therefore, combining the two networks in series, where **G** feed directly into **H**, results in a new network whose inputs and target outputs are not only known but also identical. Now, the supervised training can be applied to the combined network.

The combined network contains three hidden layers; the mapping, the bottleneck, and de-mapping layers. The second hidden layer is referred to as the *bottleneck layer* because it has the smallest dimension among the three layers. Note that the nodes in the mapping and de-mapping layers must have nonlinear transfer functions to model arbitrary **G** and **H** functions. However, nonlinear transfer functions are not necessary in the bottleneck layer. If the mapping and de-mapping layers were eliminated and only the linear bottleneck layer were left, this network would reduce to linear PCA as demonstrated by Sanger (1989). Typically M_1 and M_2 are selected to be larger than m and they are set to be equal ($M_1 = M_2$). Hereafter, the dimensions of the mapping and de-mapping layers are collectively referred to as the dimension of the mapping layers and denoted as M.

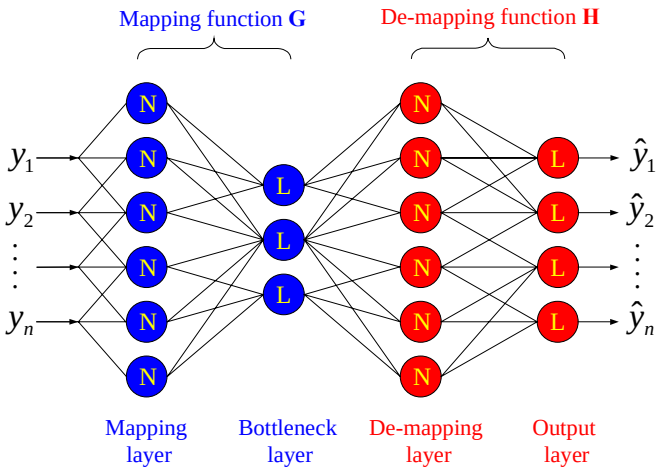

Figure 1: A schematic presentation of an auto-associative neural network

In this study, the auto-associative network is employed to reveal the latent relationship between the measured features and the unmeasured intrinsic parameters causing the variations of the measured features. For example, the measured fundamental frequency of the Alamosa Canyon Bridge in New Mexico varied approximately 5% during a 24-hour test period, and the change of the fundamental frequency was correlated to the temperature difference across the bridge deck (Sohn et al., 1999). (Because the bridge is approximately aligned in the north and south direction, there is a large temperature gradient between the west and east ends of the bridge deck throughout the day.) The auto-associative neural network presented here can be trained to learn these correlations and reveal the inherent variables driving the changes. Then, assuming that the neural network is trained to capture the embedded relationships, the prediction error of the neural network will grow when an irrelevant data set, such as ones obtained from a damage state of the system, is fed to the network. Based on this assumption, the auto-associate network is incorporated with novelty detection, which is described in the following section. The objective of novelty detection is to observe a sequence of patterns and signal if one significantly differs from the rest of population.

3. NOVELTY INDEX

The objective of the present novelty detection is to eschew the physics-based model approaches such as finite element analysis, and therefore pave the way for signal-based techniques applicable to systems of arbitrary complexity. However, the present novelty detection provides an indication only about the presence of damage in a system of interest. This method does not give information about the location and extent of the damage. That is, the novelty detection only identifies if a new pattern differs from previously obtained patterns in some significant respect. Although the damage assessment problem can be posed with several levels of complexity, the detection of damage presence is arguably the most important step. Once the existence of damage is confirmed the system can be taken out of service and subjected to detailed inspection to locate and quantify damage. The concept of novelty detection is not entirely new and applications in other fields can be found in literature (Bishop, 1994; Tarassenko et al., 2000; Worden et al., 2000).

For the current specific application of our interest, the auto-associative neural network will be trained using features extracted from the healthy baseline system and the threshold value for the novelty index will be established accordingly. When damage occurred in the system, the damage would alter the dynamic characteristics of the system and consequently the novelty indicator would signal a fault. One of the biggest challenges here is to identify significant system changes such as structural damage and degradation which cannot be attributed to natural fluctuations in the system responses due to changing environmental and operation variations. As described above, the auto-associative neural network is forced to learn the underlying dependency of the extracted features on these natural variations. Therefore, when the auto-associative network is fed with the inputs obtained from an unprecedented state of the system, for example, a damage state of the system, the novelty index (*NI*), which is defined as the Euclidean distance between the target outputs and the outputs of the neural network (Worden, 1997):

$$NI(\mathbf{y}) = \| \mathbf{y} - \hat{\mathbf{y}} \| \tag{7}$$

where $\mathbf{y}$ and $\hat{\mathbf{y}}$ are each individual columns of $\mathbf{Y}$ and $\hat{\mathbf{Y}}$ in Equation (3). If the learning has been successful, $\mathbf{y} \approx \hat{\mathbf{y}}$ and $NI(\mathbf{y}) \approx 0$ for all data in the training data set. However, if $\mathbf{y}$ were acquired after damage is introduced to the system, $NI(\mathbf{y})$ would noticeably departure from zero providing an indication of an abnormal condition of the system.

The novelty index can be also defined using the Mahalanobis distance measure between the target outputs and the network outputs (Duda and Hart, 1973):

$$NI(\mathbf{y}) = \sqrt{(\mathbf{y}-\hat{\mathbf{y}})^{\mathrm{T}}\boldsymbol{\Sigma}^{-1}(\mathbf{y}-\hat{\mathbf{y}})} \tag{8}$$

where $\boldsymbol{\Sigma}$ is the sample covariance matrix of the training data. This covariance matrix can be calculated with or without the potential outlier in the sample depending upon whether inclusive or exclusive measures are preferred (Barnett and Lewis, 1994). In this study, the first definition of the novelty index is employed.

4. EXAMPLE

4.1. Description of the Numerical Example

The proposed novelty detection technique is demonstrated using a simplified model of a computer hard disk (MathWorks, 1998). Using Newton's law, the second order differential equation for the read/write head shown in Figure 2 can be written as follows:

$$J\frac{d^2\theta}{dt^2} + C\frac{d\theta}{dt} + K\theta = K_i i \tag{9}$$

where J is the inertia of the head assembly, C is the viscous damping coefficient of the bearings, K is the return rotational spring constant, K_i is the motor torque constant, θ is the angular position of the head, and i is the input current. Although most modern hard disks have closed-loop controllers to accurately position the read/write head, reduce the seek time of the hard disk, and stabilize the system, the feedback compensator of the hard disk is omitted in this example for simplicity.

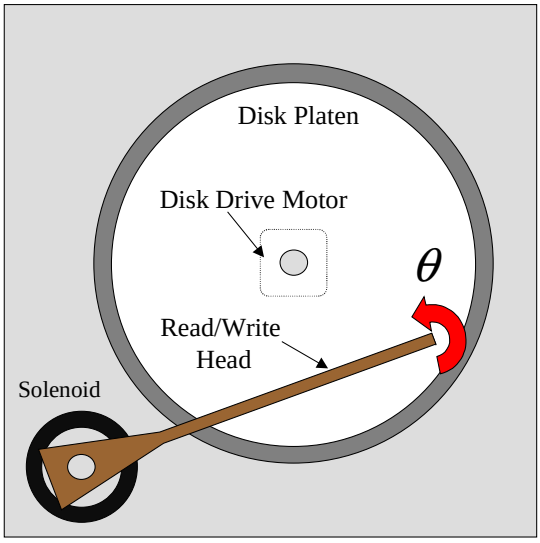

Figure 2: A computer hard disk drive

Note that although the example presented in this study is simple, the proposed method has much wider applicability than this simulation because the method presented does not assume any physics-based modeling. For instance, when detecting faults in a composite plate, the complexity of the geometry, boundary conditions, and the lay-up make it difficult to model the baseline structure. Furthermore, the modeling of damage such as fiber pullout, fiber fracture, matrix fracture, and delamination could be even more difficult (Worden, 1997). The proposed method combining the auto-associative network and novelty index only requires a sequence of measurements corresponding to the normal conditions of the system.

To simulate an operational variation of the system, it is assumed that the values of K, K_i, J, and C are a function of an ambient temperature, T, as shown in Figure 3. For example, the nominal values of K, K_i, J, and C are 10 Nm/rad, 0.047 Nm/rad, 0.01 Kg-m, 0.0 Nm/(rad/sec), respectively, at $\mathrm{T}=15\,\mathrm{C}^{\mathrm{o}}$. For the temperature range of $(-15\,\mathrm{C}^{\mathrm{o}}, 45\,\mathrm{C}^{\mathrm{o}})$, K, K_i, and J values vary about $\pm 20\%$ from this nominal values at $\mathrm{T}=15\,\mathrm{C}^{\mathrm{o}}$. C is simply changed from –0.004 to +0.004 although the negative damping value does not have any physical meaning into it. The explicit expressions for these temperature dependent variables are assigned as follows:

$$K = \frac{6}{87}(0.1\times\mathrm{T}-1.5)^3 + \frac{4}{87}(0.1\times\mathrm{T}-1.5)+10 \tag{10}$$

$$K_i = \frac{0.01}{30}\left[(0.1\times\mathrm{T}-1.5)^3 + (0.1\times\mathrm{T}-1.5)^2 + (0.1\times\mathrm{T}-1.5)+1\right] + \frac{0.14}{3} \tag{11}$$

$$J = 0.01\left(\frac{\mathrm{T}}{15}+9\right) \tag{12}$$

$$C = 0.004\times\tanh\left(\frac{\mathrm{T}}{30}\pi - \frac{\pi}{2}\right) \tag{13}$$

The temperature dependencies of these variables are arbitrarily assumed without any physical understandings of the actual system.

Taking the Laplace transform of Equation (9) and discretizing the continuous transfer function, the discrete transfer function, $H(z)$, from i to θ is obtained:

$$H(z) = \frac{b_1 z + b_2}{z^2 + a_1 z + a_2} \tag{14}$$

The coefficients of the transfer function in Equation (14) are chosen as features for the subsequent network training. Here, feature extraction refers to identifying the salient features of data to facilitate its use in a subsequent analysis, in the current case, the novelty detection. That is, features are a set of variables derived from the original data set and they are supposed to capture the relevant information contained in the original data. Because of the underlying dependencies of K, K_i, J, and C on T, a_1, a_2, b_1, and b_2 also become temperature dependent variables as shown in Figure 4. In actual applications, these coefficients should be estimated by using time series analyses (Box et al., 1994) or system identification techniques (Ljung, 1999). However, for simplicity, the coefficients for the numerical transfer function are used in this example.

The *superficial dimensionality* of data, or the number of observations, is often much larger than the *intrinsic dimensionality*, or the number of independent variable causing the underlying variations in the observations. This is also true in the current example because four parameters (a_1 , a_2 , b_1 , b_2) are extracted and there is only one intrinsic variable (T) driving the changes of these four parameters. The auto-associative neural network should be able to capture these nonlinear/linear dependencies of the transfer coefficients on the temperature.

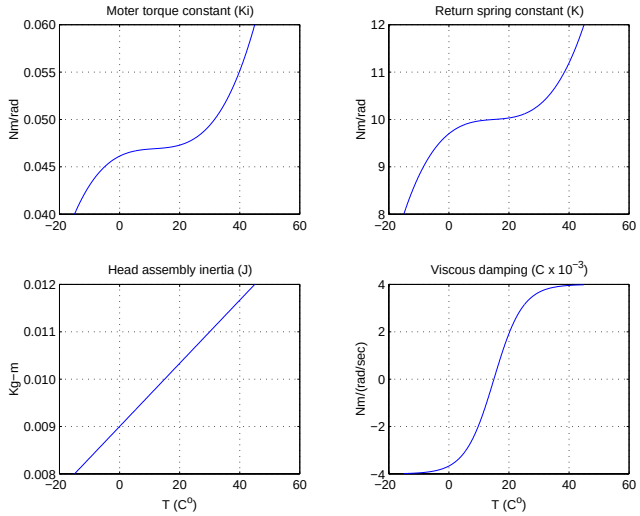

Figure 3: Temperature variation of *K*, K_i, *J*, *C*

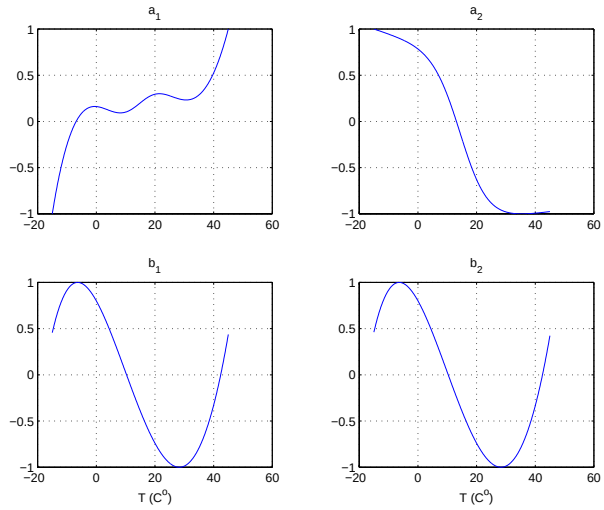

Figure 4: Temperature variation of a_1, a_2, b_1, b_2

4.2. Training of Auto-Associative Neural Network

In order to train the neural network, the coefficients of the transfer function, a_1, a_2, b_1, b_2 are specified as inputs to the auto-associated neural network. Assuming a uniform distribution of temperature in the range of ($-15\,\mathrm{C}^o$, $45\,\mathrm{C}^o$), *K*, K_i, *J*, and *C* values are computed at randomly selected 600 temperature values according Equations (10)-(13). Then, the associated a_1, a_2, b_1, b_2 coefficients are obtained, corrupted with Gaussian noises, and used as the training data set. That is, the data set consists of 600 observations with 4 input variables (*m*=4 and *n*=600). The data set was scaled so that each variable ranges from –1 to 1. This scaling weighs all four variables equally important and is similar to the division of data set by standard deviation often used in the preparation of data for PCA. It should be noted that temperature, T, is only one underlying parameter driving the changes of these coefficients. Therefore, the auto-associative neural network with only one node in the bottleneck layer should be able to reproduce this training data set (see Figure 5).

The auto-associative neural networks with different dimensions in the mapping and de-mapping layers are applied to this training data to determine the best network architecture. In general, the number of nodes in the mapping and de-mapping layers is set to be larger than that of the bottleneck layer (M_1 , $M_2 > d$). However, there are no definitive rules for deciding the dimensions of the mapping and de-mapping layers. The complexity of the nonlinear functions, which the neural network represents, primarily controls the number of nodes in the mapping and de-mapping layers. If too few nodes are specified in the mapping layers, the accuracy of the neural network might be poor. On the other hand, if too many mapping nodes are provided, the network will be prone to overfitting learning the stochastic nature of the data rather than the underlying functionalities. In practice, the available data might impose constraints on the number of nodes in the hidden layers if the number of training data sets is limited. Otherwise, explicit criteria trading off between the accuracy and the dimension of the hidden layers are often used. Two such criteria are Akaike's Final Prediction Error (FPE) and An Information theoretic Criterion (AIC) (Ljung, 1999):

$$FPE = e(1+N_t/N)/(1-N_t/N) \tag{15}$$

$$AIC = \ln[e] + 2N_t/N \tag{16}$$

where $N_t = (m+d+1)(M_1+M_2)+m+d$ is the total number of weights, $N = nm$ is the number of points in the data, $e = E/(2N)$, and E is the sum of squared errors for all entries in $\mathbf{Y}-\hat{\mathbf{Y}}$. Minimization of these criteria identifies the number of nodes that are neither underparameterized nor overfitted. In this example, a neural network with 10 nodes in each mapping and de-mapping layer has minimized the two criteria on average, and is employed for the subsequent novelty detection.

The number and time of iterations are not reported here because the iterations depend on the training method and the initial conditions. However in most cases, less than 10,000 iterations were required before convergence. Several trainings with different initial conditions were required for a given architecture to assure that the global minimum had been achieved. Also, sigmoidal transfer functions were used in all hidden layers as well as the output layer so that the outputs

were bounded in the range (-1, 1). The networks employed in this study are conventional feedforward networks and trained by a Levenberg-Marquardt version of backpropagation. It is reported that the Levenberg-Marquardt algorithm is 10 to 100 times faster than the usual gradient descent method (Hagan and Menhaj, 1994).

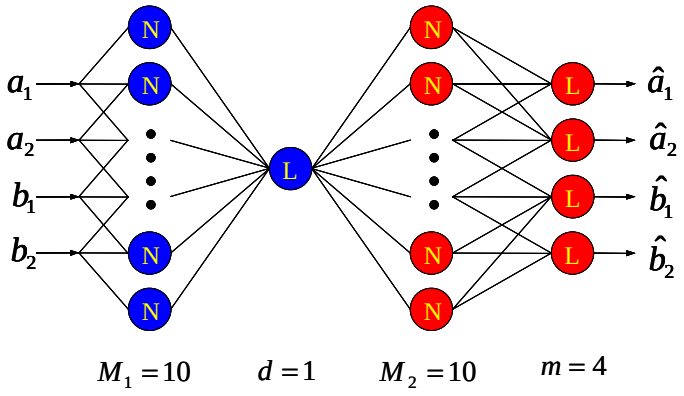

Figure 5: The neural network architecture for the hard disk drive example

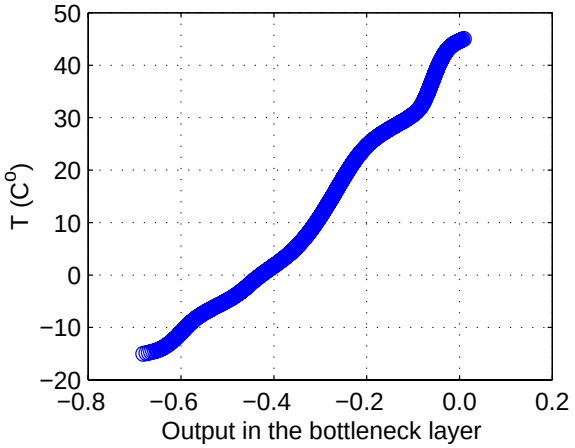

Figure 6: Correlation between temperature and the output in the bottleneck layer

Although it is not presented in this paper, the difference between the original training data **Y** and the reconstructed data $\hat{\mathbf{Y}}$ was negligible for most cases. If the neural network was successfully trained, the output of the bottleneck layer should be analogous to the unmeasured temperature T because the temperature is the only underlying intrinsic variable causing all the fluctuations. Figure 6 shows the relationship between the output of the bottleneck layer and temperature, T. The bottleneck output is indeed closely related to the temperature: the relationship, although not linear, is monotonic and this is sufficient to reconstruct the input at the output layer. Therefore, this auto-associative neural network had in a sense revealed the unmeasured temperature embedded in this data set.

4.3. Damage Scenarios

The fault in this system is simulated by changing K and C by various degrees. The four damage cases investigated in this study are summarized in Table 1. For instance, the damping coefficient of case (a) is fixed at the damping value corresponding to T= $20\,\mathrm{C}^{\circ}$ ($C_d = C_{20}$), and the damaged return spring constant, K_d, is varied between $0.85\,K_{20}$ and $0.95\,K_{20}$. Here, K_{20} is the value of the return spring constant at T= $20\,\mathrm{C}^{\circ}$. More specifically, 600 sets of K_d values are randomly sampled between $0.85\,K_{20}$ and $0.95\,K_{20}$ assuming a uniform distribution between these two values. Then, the corresponding values of a_1, a_2, b_1, and b_2 are computed, and fed to the previously trained auto-associative neural network for the computation of the novelty index. In a similar manner, input data with the size of 4×600 are generated for damage cases (b) – (d).

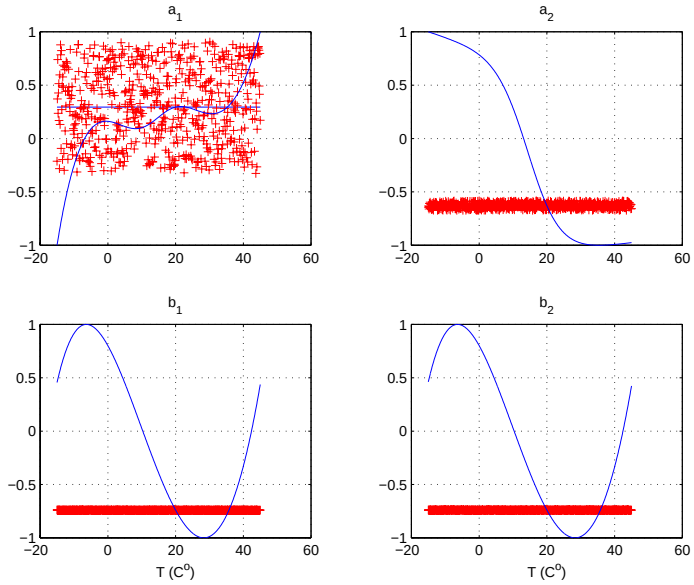

Figure 7: Comparison of the variation magnitudes caused by ambient temperature and damage case (d)
(-: variation caused by temperature, +: variation cause by damage)

To provide a perspective of the variation magnitudes caused by damage and ambient temperature, Figure 7 shows the fluctuations of the transfer function coefficients associated with damage case (d) in Table 1. It is clearly shown that, in this example, temperature produces much larger changes in these coefficients than damage. Therefore, without special cautions and treatment, it is very difficult to identify what is causing these variations. This kind of observation can be often found in many applications. For example, dynamic characteristics of offshore platforms undergo significant variations in time as a result of tides and change of oil storage producing a continuous range of normal conditions. In this case, it is clearly undesirable for the novelty detector to signal damage simply because of a

change in the environment. The presented auto-associative network can help to address this issue by learning the concealed dependency of the network inputs on the unmeasured intrinsic parameters.

Table 1: Damage scenarios investigated in this study

Cases	Spring constant (K_d)	Viscous damping (C_d)
(a)	$[0.85\,K_{20}, 0.95\,K_{20}]$	C_{20}
(b)	K_{20}	$[0.90\,C_{20}, 1.10\,C_{20}]$
(c)	$[0.95\,K_{20}, 1.05\,K_{20}]$	C_{20}
(d)	$[0.95\,K_{20}, 1.05\,K_{20}]$	$[0.90\,C_{20}, 1.10\,C_{20}]$

4.4. Novelty Detection

First, validation data corresponding to the baseline system are created in a similar way to the generation of the training data set. That is, for a randomly selected temperature value, the physical parameters and the coefficients of the transfer function are computed. Then, the auto-associative neural network takes the coefficients as inputs and computes the novelty index. This procedure is repeated 600 times to generate the same number of novelty measures.

For each damage case in Table 1, a_1, a_2, b_1, and b_2 coefficients are obtained from the partially perturbed K_d, K_i, J, and C_d. Then, the novelty index defined in Equation (7) is computed after feeding these coefficients into the previously trained network. The diagnosis results are displayed in Figure 8. Case (a) produces novelty values, which show gross changes and visual inspection suffices to identify the fault. Cases (c) and (d) result in more subtle changes but still noticeable changes. However, case (b) does not display any distinct changes.

The establishment of a threshold value can be useful to decide if "statistically significant" changes have occurred in the system condition. However, the construction of the threshold value based on a rigorous statistical analysis is not achieved in this study. Further investigation is necessary to address this issue. Worden (1997) and Cempel (1985) established the *warning level*, above which it is considered that a reading is sufficiently abnormal to require investigation. The computation of the warning level is based on a continuous adjustment of the mean and standard deviation of the parameter records, and then confidence intervals can be assigned assuming a Gaussian distribution of the records.

Based on permutation theory, Box and Andersen (1955) proposed a modified hypothesis test to safely use in more general applications without a normality assumption. The primary objective is to test the null hypothesis, H_0: $\sigma^2(x) = \sigma^2(y)$ against the one-sided alternative H_1: $\sigma^2(x) < \sigma^2(y)$. Here $\sigma^2(x)$ and $\sigma^2(y)$ are the variances of arbitrary variables x and y, respectively. This modified hypothesis test can be employed to check if the new signal has significantly changed from the training data set. Various studies based on Monte Carlo simulation (Miller 1997 and references therein) have demonstrated that this Box-Andersen test maintains reasonably correct significant levels under the null hypothesis for a variety of heavy- and short-tailed distributions.

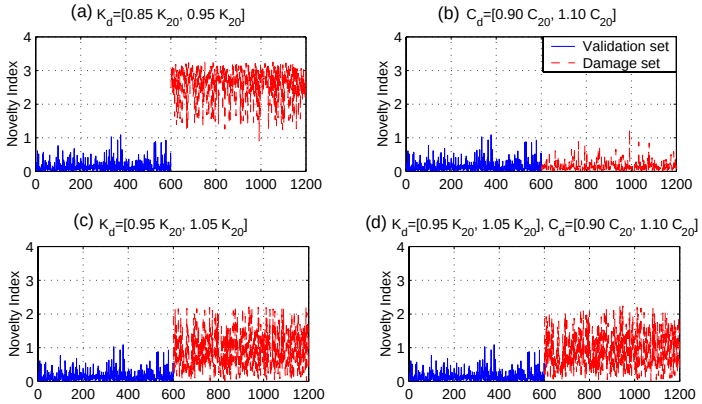

Figure 8: Novelty indices evaluated at four different damage cases

5. CONCLUSIONS

This paper extends the previous work on novelty detection for structural damage diagnosis, explicitly taking into account changing environmental and operational conditions. An attempt is made to discriminate the changes of system responses due to ambient operational conditions from those caused by structural damage. The proposed approach is demonstrated using a simplified model of a computer hard disk. Results indicate that the incorporation of the auto-associative network with novelty measure enables to detect damage even when the system exhibits a rage of normal conditions. The development presented here may allow some progress in in-service monitoring of aerospace, automobile, civil, and mechanical systems, which are subject to various operational and environmental conditions.

Before the proposed approach could be used with confidence on experimental data, several issues need to be addressed. First, this study assumes that environmental variations and damage have uncorrelated effects on the system's response, making it easier to discriminate them. In some cases, the environmental effects, however, have similar influence on the system's dynamic characteristics as damage has. In this case, the discrimination between environmental effects and damage becomes more difficult. Therefore, further studies are needed for this situation.

Second, the sensitivity of the novelty index performance based on different noise types and levels need to be further investigated. It is also important to establish what degree of changes in the novelty is statistically significant. Joint research work is currently underway at Los Alamos National Laboratory and University of Sheffield to address these issues.

REFERENCES

1. V. Barnett and T. Lewis, *Outliers in Statistical Data*, John Wiley & Sons, Chichester, 1994.
2. C. M. Bishop, "Novelty detection and neural network validation," *IEE Proceedings-Vision and Image Processing*, **141**, pp. 217-222, 1994.
3. C. M. Bishop, *Neural Networks for Pattern Recognition*, Oxford University Press, Oxford, UK, 1995.
4. G. E. P. Box, and S. L. Andersen, "Permutation theory in the derivation of robust criteria and the study of departures from assumption," *Journal of the Royal Statistical Society*, Series **B17**, pp. 1-26, 1955.
5. G. E. P. Box, G. M. Jenkins, and G. C. Reinsel, *Time Series Analysis: Forecasting and Control*, Third Edition, Prentice-Hall, Inc., New Jersey, 1994.
6. G. Cybenko, "Approximation by superposition of a signoidal function," Math. Control Signal & System, **2**(4), pp. 303-314, 1989.
7. R. O. Duda, and P. E. Hart, Pattern Classification and Scene Analysis, John Wiley & Sons, New York, 1973.
8. K. Fukunaga and W. L. G. Koontz, "Application of Karhunen-Loeve Expansion to Feature Selection and Ordering," *IEEE Transactions on Computers*, **C-19** (4), pp. 311-318, 1970.
9. K. Fukunaga, *Statistical Pattern Recognition*, Academic Press, San Diego, CA, 1990.
10. M. T. Hagan, and M. Menhaj, "Training Feedforward Networks with the Marquardt Algorithm," *IEEE Transactions on Neural Networks*, **5**(6), pp. 989-993, 1994.
11. M. A. Kramer, "Nonlinear Principal Component Analysis Using Autoassociative Neural Networks," *AIChE Journal*, **37**, pp. 233-243, 1991.
12. L. Ljung, *System Identification-Theory for the User*, Prentice Hall, Upper Saddle River, New Jersey, 1999.
13. MathWorks, Inc., Control System Toolbox User's Guide, MathWorks, Inc., 1998.
14. R. G. Miller, *Beyond ANOVA: Basics of Applied Statistics*, Chapman&Hall/CRC, New York, 1997.
15. D. E. Rumelhart and J. L. McClelland, *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*, MIT Press, Cambridge, MA, 1988.
16. T. D. Sanger, "Optimal Unsupervised Learning in a Single-Layer Linear Feedforward Neural Network," Neural Networks, **2**(6), pp.459-473, 1989.
17. H. Sohn, M. Dzwonczyk, E. G. Straser, A. S. Kiremidjian, K. H. Law, and T. Meng, "An Experimental Study of Temperature Effects on Modal Parameters of the Alamosa Canyon Bridge," *Earthquake Engineering and Structural Dynamics*, **28**, pp. 879-897,1999.
18. L. Tarassenko, A. Nairac, N. Townsend, I. Buxton, Z. Cowley, "Novelty Detection for the Identification of Abnormalities," *International Journal of Systems Science*, **31**(11), pp.1427-1439, 2000.
19. K. Worden, "Structural Fault Detection Using A Novelty Measure," *Journal of Sound and Vibration*, **201**(1), pp. 85-101, 1997.
20. K. Worden, G. Manson, and N. R. J. Fieller, "Damage Detection Using Outlier Analysis," *Journal of Sound and Vibration*, **229**(3), pp. 647-667, 2000.